



Designing a multi-depot multi-period vehicle routing problem with time window: hybridization of tabu search and variable neighbourhood search algorithm

ARJUN PAUL¹, RAVI SHANKAR KUMAR², CHAYANIKA ROUT¹ and ADRIJIT GOSWAMI^{1,*}

¹Department of Mathematics, Indian Institute of Technology Kharagpur, Kharagpur 721302, India

²Government Polytechnic, Saharsa, DST Bihar, Bihar 852201, India

e-mail: 1729arjun@gmail.com; ravikhushi412@yahoo.co.in; fantu.rout@gmail.com;

goswami@maths.iitkgp.ac.in

MS received 18 December 2019; revised 8 July 2020; accepted 20 July 2021

Abstract. This research formulates a real-life multi-depot and multi-period vehicle routing problem (MDMPVRP) by imposing time window (TW) and many other constraints. A set of customers, spread in different locations, are to be served by a fleet of heterogeneous vehicles over a finite number of periods. Each customer is associated with combinations of routes and vehicles over the period. A customer must be served in one of the allowable combinations. The objective of this MDMPVRP-TW is to minimize the total distance traversed by the fleet over the planning horizon. The proposed MDMPVRP-TW is an extension of vehicle routing problem (VRP), and is hence an NP-hard problem. In order to optimize it, we propose a hybrid meta-heuristic approach by combining tabu search (TS) and variable neighbourhood search (VNS) algorithms. Furthermore, to provide richer insights, the efficacy of the proposed method and mathematical formulation is demonstrated through numerical experiments for a number of instances varying from small to large scale.

Keywords. Multi-depot vehicle routing problem; multi-period; time window; allowable combination; hybrid VNS-TS.

1. Introduction

Vehicle routing problem (VRP) determines a set of routes for a fleet of vehicles aiming to minimize, in general, the total distance traversed under a set of constraints such as capacity, time window (TW), visiting frequency, etc. VRP with time window (VRP-TW) imposes a constraint on the start of service time, thereby indicating that a customer must be visited by a vehicle within prescribed time interval. In order to encounter such real-life business scenarios, VRP-TW is observed to arise frequently in various context such as grocery distribution, oil and petroleum delivery, repairmen scheduling, school bus routing, industrial refuse collection, etc. (Cordeau et al [1]). The complexity involved in VRP-TW is much higher when the fleet is assumed to consist of a set of heterogeneous vehicles (having different capacities), which traverse through multiple depots over multiple periods. Yousefikhoshbakht *et al* [2] proposed a meta-heuristic algorithm called bone route algorithm combined with a modified tabu search (BRMTS) for solving the heterogeneous fixed fleet open vehicle

routing problem (HFFOVRP). The proposed research is intended to formulate a multi-depot multi-period vehicle routing problem with time window (MDMPVRP-TW), wherein a transportation company builds a set of replenishment depots to serve a set of customers located at different places. A fleet of heterogeneous vehicles are operated over a finite time horizon, the latter being segmented into several periods. The proposed MDMPVRP-TW is an extension of VRP-TW, and is therefore an NP-hard problem. To optimize the MDMPVRP-TW, we propose a hybrid meta-heuristic by combining tabu search (TS) with variable neighbourhood search (VNS) algorithm.

Over the last few decades, VRP-TW is being widely studied and is found to have many applications in the design and management of distribution systems. Several variants of VRP-TW are therefore put forward to encounter the real-life situations more closely. Besides the capacity constraint, it is now mandatory for each vehicle to visit a delivery location within a specified time frame. In literature two types of TWs are considered, namely soft and hard. The soft TW can be violated by incurring penalty cost whereas hard TW cannot be violated at any cost. The latter case is being considered in the present work. A heuristic TS

*For correspondence

algorithm was studied by Taillard *et al* [3] with soft TW. Cordeau *et al* [1] presented two important generalizations, namely the periodic and the multi-depot vehicle routing problems (MDVRPs) with TWs. Recently, Paul *et al* [4] investigated a two-echelon pollution routing problem with simultaneous pickup and delivery under multiple TWs constraint. Some other notable contributions on VRP-TW include the works of Solomon [5], Braysy and Gendreau [6], Chang *et al* [7], Miranda and Conceicao [8], Chu *et al* [9] and Hu *et al* [10].

In real-life VRP, routes are designed for a planning horizon that may consist of multiple periods and a customer may be served over a subset of these periods (Mancini [11]). Multi-period vehicle routing problem (MPVRP) is a profitable tour planning that allows vehicles to serve the customers based on their requirements so that every customer need not necessarily be visited daily. MPVRPs, capturing different situations, have been formulated by many researchers in the recent years. Dayarian *et al* [12] proposed partitioning-based two-stage MPVRP in which first stage corresponds to plan whereas second stage takes multi-period. Luo *et al* [13] investigated a mixed integer MPVRP with TW and limited visiting quota, and proposed three-stage heuristic approach to optimize it. Dayarian *et al* [14] formulated MPVRP by considering seasonal fluctuation in supply and designed an adaptive large neighbourhood search (ALNS)-based meta-heuristic to optimize the proposed model. Naccache *et al* [15] addressed a multi-pickup and delivery VRP with TW, and developed a hybrid ALNS via branch and bound heuristic. Similar to MPVRP, the literature of MDVRP is also very rich. An elaborative literature survey on MDVRP is carried out by Lahyani *et al* [16]. Michallet *et al* [17] addressed a real-world problem occurring in the transportation of valuable goods by considering a very hard periodic VRP with TWs and time spread constraints on services. The existing MDVRPs are mostly optimized by heuristics approach. One of the first algorithms is put forward by Tillman [18], which assigns every customer node to its nearest depot and thereby routes are constructed back and forth between the depots and the customers. Renaud *et al* [19] discussed a TS algorithm for the MDVRP with capacity and route length constraints, the algorithm being tested on a set of 23 benchmark instances.

During serving the customers along a route, it may sometimes happen that the remaining cargo in the vehicle is not sufficient enough to fulfil the demand of the next customer. In such a situation there is a need of installation of some intermediate replenishment depots, which can be visited by the vehicles as and when required. Cordeau *et al* [1] presented a unified TS heuristic for the VRP-TW as well as for its two important generalizations, namely, PVRP-TW and MDVRP-TW. Allowable combination for serving the customers was also taken into consideration, which restricted the service time to be scheduled on certain specified time periods. Suppose a planning horizon of 5 days is considered, viz. Monday to Friday. Within this

planning horizon a customer i is to be served twice, with the service being provided on Monday and Wednesday, or Monday and Friday or Tuesday and Friday (say). Therefore, in this case the customer i has 3 possible combinations of 2 days so that the set of allowable combinations can be represented as $\{\{\text{Monday, Wednesday}\}, \{\text{Monday, Friday}\}, \{\text{Tuesday, Friday}\}\}$. Polacek *et al* [20] proposed an algorithm based on the philosophy of the VNS for solving MDVRP-TW. An extension of MDVRP was suggested by Crevier *et al* [21] involving replenishment of vehicles at intermediate depots along the routes. A three-phase methodology was proposed by them based on adaptive memory, TS and integer programming. The algorithm was tested on randomly generated as well as on benchmark instances derived from those proposed for MDVRP by Cordeau *et al* [22]. A hybrid granular tabu search algorithm to solve the MDVRP under associated capacity and maximum duration constraints was proposed by Escobar *et al* [23].

Among the recent works, Mancini [11] introduced the multi-depot multi-period vehicle routing problem with a heterogeneous fleet (MDMPVRPHF). The authors proposed a mixed integer programming formulation and an ALNS-based meta-heuristic that defines several destroy operators. Salhi *et al* [24] solved an MDVRP with heterogeneous vehicle fleet by designing an efficient VNS-based algorithm. Kumar *et al* [25] developed a multi-objective multi-vehicle production and pollution routing problems with time window (MMPPRP-TW). They formalized a hybrid self-learning particle swarm optimization (SLPSO) algorithm in multi-objective framework for the solution methodology. A multi-depot open VRP was investigated by Soto *et al* [26], which is a well-known variant of VRP in which vehicles are not required to return to the depot at the end of their tour. Instead, each route ends at the last served customer. Recently, Zhou *et al* [27] introduced a new city logistics problem by considering a multi-depot two-echelon VRP with delivery options. For the solution methodology of the problem, the authors proposed a hybrid multi-population genetic algorithm.

With an overview of the related research works done so far, it is observed that a VRP-TW model with both multi-period and multi-depot facilities considering allowable combinations has not been examined yet in the literature. Therefore, in order to fill this research gap, a model with the said characteristics is presented here. The second novelty of this research is to propose a hybrid meta-heuristic by combining TS and VNS. The remainder of the paper is organized as follows. A detailed description of the examined situation with the formulation of the mathematical model is demonstrated in Section 2. The proposed hybrid VNS-TS algorithm is elaborately discussed in Section 3. In Section 4, the proposed model is illustrated with the help of numerical examples followed by relevant discussions. Lastly, Section 5 presents a brief conclusion along with future research opportunities.

2. Model formulation

In this section, we mathematically formulate the proposed MDMPVRP-TW. Before this, a detailed description of the model along with real-life application is presented.

2.1 Situation description

In real-life problems, a transportation company plans to design routes to serve customers over a planning horizon instead of a single period. We have considered, in our study, a finite planning horizon subdivided into a number of periods. For instance, a week may be regarded as a planning horizon with each of the seven days being considered as a period. A customer is served over a subset of this planning horizon. For example, one may be served on Monday, Wednesday and Friday, while another is served on Saturday and Wednesday and so on. Such a planning of routes also helps in minimization of total traversed distance. Every customer i has a constant demand and a fixed service time indicating that a vehicle must visit it within the specified TW, thereby taking care of the allowable combinations as discussed in Section 1. A set of d depots are installed at various locations, where a fleet of r heterogeneous vehicles are considered to be located. The journey of a vehicle in period t starts from the depot at which its journey ended in period $(t - 1)$. While serving the customers along the route, it may happen sometimes that a vehicle is out of cargo or the remaining cargo is not sufficient to serve the next customer node. In such a situation, it may visit any nearby replenishment depot for the purpose of refilling. However, the route duration covered by any vehicle is not allowed to exceed a given maximum time limit.

As an instance, water distribution to drought-affected areas can be considered as a suitable example for the proposed MDMPVRP-TW. All the vehicles that are used for transportation purpose are situated at a main station from where tours have to begin. The various sites away from the main station are affected by drought. In such a scenario, the more affected areas are supposed to be served frequently whereas the lesser affected regions have less frequent supply for water. Accordingly, each area will be specified with an allowable combinations set within a certain planning horizon in order to counteract the water scarcity in every region equally. Moreover, at any point of service, if the water supplying tank is found to lack sufficient water to serve further when there is already some pending demand, then in such a situation the vehicle (carrying storage tanks) can visit some nearby water reservoir, get refilled and resume the service further.

2.2 Assumptions

The model is built up under the following assumptions listed:

1. Heterogeneous fleet of vehicles is considered to serve the customer.

2. Demand rate at each customer node is fixed and known in advance. Shortages of item are not permissible to any node.
3. Intermediate replenishment is considered, which means any running vehicle can be replenished by a nearby depot if required.
4. Demand of any customer does not exceed the vehicle capacity.
5. Demand at each depot is assumed to be zero.
6. The journey of a vehicle in period t starts from the depot at which its journey ended in period $t - 1$.
7. Every depot is assumed to hold infinite inventory.
8. Each customer has a TW in each period within which it must be served by at most one vehicle.
9. Any route of a vehicle will get terminated as soon as it violates maximum time limit constraint.
10. A route is a path travelled by a vehicle in any particular period.

2.3 Mathematical model

Built upon the discussed situation, a constrained optimization problem is formulated with the objective function being represented as follows:

$$\text{minimize: } \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^r \sum_{t=1}^p d_{ij} x_{ijkt}, \quad (1)$$

$$i \neq j$$

Equation (1) represents the total distance travelled by all the vehicles over the planning horizon. Hence our objective is to minimize it. The following constraints are enunciated by addressing time window, visiting frequency quota and vehicle routes:

$$e_{it} \leq ar_{ikt} \leq l_{it}, \quad \forall i \in V, \quad \forall k \in K, \quad \forall t \in T \quad (2)$$

$$ar_{ikt} + s_{ikt} + t_{ij} x_{ijkt} \leq ar_{jkt} x_{ijkt} + \Lambda(1 - x_{ijkt}), \quad \forall i, j \in V, \quad i \neq j, \quad \forall k \in K, \quad \forall t \in T \quad (3)$$

$$y_{ikt} \geq q_{it}, \quad \forall i \in V_1, \quad \forall k \in K, \quad \forall t \in T \quad (4)$$

$$0 \leq y_{jkt} \leq (y_{ikt} - q_{it}) x_{ijkt} + Q_k(1 - x_{ijkt}), \quad \forall i \in V_1, \quad \forall j \in V, \quad \forall k \in K, \quad \forall t \in T \quad (5)$$

$$0 \leq y_{jkt} \leq Q_k - q_{it} x_{ijkt}, \quad \forall i \in V_2, \quad \forall j \in V, \quad \forall k \in K, \quad \forall t \in T \quad (6)$$

$$\sum_{i \in V} \sum_{\substack{j \in V \\ i \neq j}} q_{it} x_{ijkt} \leq Q_k \left(\sum_{i \in V_2} E_{ikt} + 1 \right), \quad \forall k \in K, \quad \forall t \in T \quad (7)$$

$$t_{ij} = \frac{d_{ij}}{v}, \quad \forall i, j \in V$$

$$\sum_{i \in V} \sum_{\substack{j \in V \\ i \neq j}} t_{ij} x_{ijkt} \leq D_k, \quad \forall k \in K, \quad \forall t \in T$$

$$\sum_{j \in V} \sum_{k \in K} x_{ijkt} \leq 1, \quad \forall i \in V_1, \quad \forall t \in T, \quad i \neq j \quad (10)$$

$$\sum_{j \in V} \sum_{k \in K} x_{jikt} \leq 1, \quad \forall i \in V_1, \quad \forall t \in T, \quad i \neq j \quad (11)$$

$$\sum_{j \in V} x_{ijkt} = \sum_{j \in V} x_{jikt} \quad \forall i \in V_1, \quad \forall k \in K, \quad \forall t \in T \quad (12)$$

$$\sum_{j \in V} \sum_{k \in K} \sum_{t \in T} x_{ijkt} = f_i \quad \forall i \in V_1 \quad (13)$$

$$\sum_{j \in V} \sum_{k \in K} \sum_{t \in T} x_{jikt} = f_i \quad \forall i \in V_1 \quad (14)$$

$$\sum_{r \in C_i} w_{ir} = 1, \quad \forall i \in V_1 \quad (15)$$

$$\sum_{j \in V_1} \sum_{k \in K} x_{ijkt} - \sum_{r \in C_i} a_{rt} w_{ir} = 0, \quad \forall i \in V_1, \quad \forall k \in K, \quad i \neq j, \quad \forall t \in T \quad (16)$$

$$\sum_{j \in V} \sum_{k \in K} x_{ijkt} = \sum_{k \in K} E_{ikt}, \quad \forall i \in V_2, \quad \forall t \in T \quad (17)$$

$$\sum_{j \in V} \sum_{k \in K} x_{jikt} = \sum_{k \in K} E_{ikt}, \quad \forall i \in V_2, \quad \forall t \in T \quad (18)$$

$$\sum_{k \in K} M_{ikt} \leq r, \quad \forall i \in V_2, \quad \forall t \in T \quad (19)$$

$$\sum_{k \in K} N_{ikt} \leq r, \quad \forall i \in V_2, \quad \forall t \in T \quad (20)$$

$$\sum_{i \in V_2} \sum_{k \in K} M_{ikt} = \sum_{i \in V_2} \sum_{k \in K} N_{ikt}, \quad \forall i \in V_2, \quad \forall t \in T \quad (21)$$

$$\sum_{i \in V_2} M_{ikt} + \sum_{i \in V_2} E_{ikt} + \sum_{i \in V_2} N_{ikt} \geq 2, \quad \forall k \in K, \quad \forall t \in T \quad (22)$$

$$M_{ikt} = N_{ik,t-1}, \quad \forall i \in V_2, \quad \forall k \in K, \quad \forall t \in T \setminus \{1\} \quad (23)$$

$$\sum_{i \in S} \sum_{\substack{j \in S \\ i \neq j}} x_{ijkt} \leq \sum_{i \in S} \sum_{\substack{j \in V_1 \\ i \neq j}} x_{ijkt} - \sum_{\substack{j \in V_1 \\ e \neq j}} x_{ejkt}, \quad \forall S \subseteq V_1, \quad |S| \geq 2, \quad \forall e \in S, \quad \forall k \in K, \quad \forall t \in T \quad (24)$$

$$a_{rkt} \geq 0, \quad \forall i \in V, \quad \forall k \in K, \quad \forall t \in T \quad (25)$$

$$y_{ikt} \geq 0, \quad \forall i \in V, \quad \forall k \in K, \quad \forall t \in T \quad (26)$$

$$w_{ir} \in \{0, 1\}, \quad \forall i \in V_1, \quad \forall r \in C_i \quad (27)$$

$$E_{ikt} \in \{0, 1\}, \quad \forall i \in V_2, \quad \forall k \in K, \quad \forall t \in T \quad (28)$$

$$M_{ikt} \in \{0, 1\}, \quad \forall i \in V_2, \quad \forall k \in K, \quad \forall t \in T \quad (29)$$

$$N_{ikt} \in \{0, 1\}, \quad \forall i \in V_2, \quad \forall k \in K, \quad \forall t \in T \quad (30)$$

$$x_{ijkt} \in \{0, 1\}, \quad \forall i, j \in V, \quad \forall k \in K, \quad \forall t \in T \quad (31)$$

$$x_{iikt} = 0, \quad \forall i \in V, \quad \forall k \in K, \quad \forall t \in T. \quad (32)$$

TW constraints are given in (2) and (3), which indicate that a vehicle has to reach a particular customer node within a given time interval. Constraints (4) ensure that the remaining cargo in any vehicle on arrival at a node is greater than the customer demand at that node. It is specified by constraints (5) and (6) that the amount of cargo left in a vehicle can never fall below zero, i.e., whenever required, it may visit any nearby intermediate depot along the route for replenishment. Constraints (7) put a restriction on the total load carried by a vehicle, ensuring that it cannot exceed the vehicle capacity. Relation between time and velocity is given by (8). Constraints (9) restrict the duration of a route to its maximum limit. It is ensured by constraints (10) and (11) that in each time period, a customer can be served by at most one vehicle. Constraints (12) indicate that a vehicle arriving at a customer node will also get departed. Constraints (13) and (14) state that a customer i will be altogether served f_i times over the planning horizon. It is guaranteed by constraints (15) and (16) that each customer is visited only on the days corresponding to the assigned combination. Constraints represented by these two equations are directly adopted from the original research work of Cordeau *et al* [22]. Constraints (17) and (18) confirm the departure of a vehicle from an intermediate depot in case if it has visited one. A restriction upon the total number of

vehicles is imposed by constraints (19) and (20). All vehicles departing from various depots in a particular period are bound to return back to the depots, which is ensured by constraints (21) and (22). Constraints (23) suggest that the journey of any vehicle in period t starts from the depot where it ended in period $(t - 1)$. For each vehicle the possibility of sub-tours is eliminated by constraints (24), as discussed in the work of Kumar *et al* [25]. Constraints (25)–(32) define the domain of the decision variables.

3. Solution methodology

As discussed in introduction section, the literature is laden with several algorithms for different variants of VRP and also with various ways to obtain the initial solution. The algorithm proposed in this paper, to obtain an optimal solution to the formulated MDMPVRP-TW, is a hybridization of TS and VNS. A detailed description of the adopted procedure is presented under the following headings.

3.1 Initial solution

The proposed approach to construct the initial solution is quite similar to that suggested by Cordeau *et al* [1], and is a very fast method. Step-by-step description of the procedure is listed as follows:

1. For each customer, an allowable combination from the given options is randomly selected and allotted.
2. Sets of customers having demand (need a service) in the respective periods are to be listed.
3. For every period:
 - (a) Customers are sorted in ascending order of the mean value of their corresponding TW and the same is stored in a set U .
 - (b) r empty routes are created, where empty route indicates a route containing two depots.
 - (c) From the first α customers of U , a customer i is randomly selected and inserted into any one of the r routes so that it is feasible with respect to TW and route duration. Then, i is removed from U .
 - (d) Step (c) will be repeated until the remaining customers of U cannot be inserted into any route without violating TW and maximum duration constraints and then the remaining customers of U are inserted into the r^{th} route.
 - (e) If the remaining cargo in a vehicle after serving a customer is not sufficient to serve the next customer along the route, then a depot is inserted in between these two customers in a way so as to minimize the total distance.

Thus, a set of r routes for each period is obtained where the first $(r - 1)$ routes are feasible with respect to demand (TW and route duration constraints may be violated because of late insertion of the intermediate depot in the routes). After completing this process we have an initial solution.

3.2 Generalized cost function

A solution is evaluated by means of the following generalized cost function:

$$f(s) = f_{dist}(s) + \alpha f_{dem}(s) + \beta f_{dur}(s) + \gamma f_{wait}(s) + \delta f_{late}(s). \quad (33)$$

The terms used in the relation have the following significances: $f_{dist}(s)$ denotes the total distance covered by all the routes in the entire planning horizon, $f_{dem}(s)$ represents the demand penalty, $f_{dur}(s)$ is the route duration penalty, $f_{wait}(s)$ denotes the total waiting time penalty, $f_{late}(s)$ is delayed time penalty and $\alpha, \beta, \gamma, \delta$ are the penalty factors, which are considered to be constants (for details, see Schneider *et al* [28]).

If the remaining cargo is calculated to be negative at any node, then it implies that the demand of the previous node cannot be fulfilled by the available inventory on that vehicle. This will incur some demand penalty. Considering all routes in all the periods, the overall demand penalty is calculated as

$$f_{dem}(s) = \sum_{i,k,t} \max\{-y_{ikt}, 0\}. \quad (34)$$

The running time of vehicle k can never exceed a given maximum duration D_k time units, violating which will incur the duration penalty. The total duration penalty for all the periods can be given by

$$f_{dur}(s) = \sum_{k,t} \max\{dur_{kt} - D_k, 0\} \quad (35)$$

where dur_{kt} denotes the duration of route k in period t .

If a vehicle arrives at a customer before the starting time of service then it will incur a wait penalty, whereas a late penalty will be incurred if it arrives after the end of TW of service; these two can be, respectively, expressed as follows:

$$f_{wait}(s) = \sum_{i,k,t} \max\{e_{it} - ar_{ikt}, 0\} \text{ and} \quad (36)$$

$$f_{late}(s) = \sum_{i,k,t} \max\{ar_{ikt} - l_{it}, 0\}. \quad (37)$$

3.3 Proposed algorithm

The proposed MDMPVRP-TW is an NP-hard problem with heterogeneous vehicle, multiple depots and multiple

periods, wherein a customer will be served (if served in a period) based on pre-specified allowable combination. Hence this is a very complex problem, and the existing meta-heuristic approaches are not so much suitable to optimize it. Here we propose a hybrid VNS algorithm by combining TS, and call it VNS–TS algorithm. The proposed VNS–TS algorithm strengthens the neighbourhood structure of classical VNS, which will be discussed in next and onward paragraphs. In VNS algorithm, a set of neighbourhood structures $\mathcal{N}_\kappa(S)$, $\kappa = 1, \dots, \kappa_{max}$ is generated for the current best solution S . A random point S' is selected from $\mathcal{N}_\kappa(S)$ on which local search is applied to obtain a new point S'' . If the obtained solution is found to be better than the incumbent, then S is replaced by S'' and the search is thereby continued by setting $\kappa = 1$; otherwise κ is incremented by unity.

The proposed algorithm largely differs from those present in the literature in various aspects, mainly in the procedure of constructing the neighbourhood structures for a given solution. The neighbourhood structure of VNS is modified into a two-step procedure, and then local search is also utilized in the proposed VNS–TS. A brief sketch of the algorithm with detailed characteristics is presented here.

Algorithm: A hybrid meta-heuristic for MDMPVRP-TW

```

S ← Generate an initial solution()
set  $\kappa = 1$ 
set  $TC(i) = 0$  for  $i = 1, 2, \dots, n$ 
repeat
→ generate  $\mathcal{N}_\kappa(S)$ , neighbourhood structure of  $S$ 
→ update  $TC(i)$ 
→ randomly select a set  $\mathcal{S}$  of  $\theta$  solutions from  $\mathcal{N}_\kappa(S)$ 
→ for each solution  $s \in \mathcal{S}$ , generate  $\mathcal{N}'(s)$ 
→ from each  $\mathcal{N}'(s)$ ,  $s \in \mathcal{S}$ , randomly select a solution  $s'$  and store it in  $S'$ 
→ apply local search on each  $s' \in S'$ 
→ find the best solution  $S''$  from  $S'$ 
→ if  $f(S'')$  is better than  $f(S)$ 
→    $S = S''$ 
→    $\kappa = 1$ 
→ else
→    $\kappa = \kappa + 1$ 
→ endif
until stopping criterion is met

```

The proposed κ -neighbourhood structure $\mathcal{N}_\kappa(S)$ of S is generated as follows. To diversify the search, tabu counter $TC(i)$ is set for each customer i whose value initially is taken to be zero. If a customer i participates in the nbd-structure then $TC(i)$ will be set as ω , which implies it will not participate in nbd-structure for the next ω iterations. Randomly select any κ customers from a set of those having more than one allowable combination and with $TC(i)$ equal to zero. Consider all possible combinations

$\{r'_1, r'_2, \dots, r'_\kappa\}$ of their corresponding visit combinations excluding the one $\{r_1, r_2, \dots, r_\kappa\}$ present in the current solution (i.e., \exists at least one i for which $r_i \neq r'_i$). The composition of the neighbourhood structure is based upon the idea presented in the work of Cordeau *et al* [22]. It constitutes of a set of solutions that can be generated by the application of the following transformation:

For each time period l ,

if $a_{irl} = 1$ and $a_{i'r'l} = 0$ for every selected customer i in period l , then remove customer i from the routes in period l ,

if $a_{irl} = 0$ and $a_{i'r'l} = 1$ for every selected customer i in period l , then insert customer i into a route in period l so as to minimize the total function value as described in (33), where $a_{irl} = 1$ implies that, for r^{th} combination, customer i will be served on day l , and will not be served if $a_{i'r'l} = 0$.

The tabu counter for each of the selected customers is updated and that for each of the remaining nodes not participating in the nbd-structure is set as $TC(i) = \max\{0, TC(i) - 1\}$. The new set of solutions obtained at the end of this procedure computed upon all the aforesaid combinations is said to form the neighbourhood structure $\mathcal{N}_\kappa(S)$ for the current solution S . From $\mathcal{N}_\kappa(S)$, a random set \mathcal{S} of θ solutions is chosen. For each solution $s \in \mathcal{S}$, a neighbourhood structure $\mathcal{N}'(s)$ is formulated using 2-opt* operator (Potvin and Rousseau [29]). It operates on two randomly chosen routes by breaking them at certain nodes (say i and j) so as to mutually divert the portions of the routes, i.e. node i of one route gets diverted to connect with the latter part of the second selected route cut at j .

For local search three operators are used, namely, Exchange and Relocate operators (Savelsbergh [30]) and a problem-specific operator DepotInOut. They are briefly discussed here.

Exchange: The exchange operator can be utilized to swap any two customers either within the same (intra) or two different (inter) routes. No depot is allowed to participate in this operation, i.e. swapping of a depot with a customer or any other depot is prohibited.

Relocate: The relocate operator selects a random customer, removes it from the route and then inserts it into either a different position along the same route (intra) or a different route (inter). Just like the exchange operator, the relocate operator also cannot be performed on depots other than the customers.

DepotInOut: This operator finds out the first customer node at which remaining cargo becomes negative and thereby inserts a depot along the route before reaching that customer so as to minimize $f(s)$. If even after removing an intermediate depot from a route the remaining cargo in a vehicle is found to be sufficient to serve all the upcoming customers along the route without the incur of any demand penalty, then this operator removes the said depot.

For better exploration of the solution space the dynamic diversification process was adopted for the penalty factor as introduced by Cordeau *et al* [1], which was further modified by Schneider *et al* [28]. The diversification method involves updating the initial penalty factors $(\alpha_0, \beta_0, \gamma_0, \delta_0)$ with the help of a penalty updating factor λ within a permissible range of their corresponding minimum $(\alpha_{min}, \beta_{min}, \gamma_{min}, \delta_{min})$ and maximum $(\alpha_{max}, \beta_{max}, \gamma_{max}, \delta_{max})$ values.

4. Computational results

The proposed MDMPVRP-TW and hybrid VNS–TS algorithms are here tested on a set of 20 benchmark instances varying from small to large scale as presented in table 1. All the test instances are taken from Cordeau *et al* [1] after some modifications. We change the co-ordinates of the customer nodes, service time and demand parameters, and install replenishment depots randomly. All data are created in a way so that the total demand of all the customers in all the periods is greater than the total capacity of all the vehicles considered over all the periods. Therefore, the vehicles have to visit intermediate depots in order to fulfil the complete demand of the customers. In table 1, n = the number of customers, d = number of replenishment depots, m = number of available vehicles, p = total number of periods, $\sum Q_k$ = fleet capacity and $\sum q_{it}$ = total demand of all the customers for all the periods.

The parameters involved in the proposed algorithm need to be empirically selected with utmost care. This is because the choice of these parameters has a significant effect on the

Table 1. Details of instances.

Instance no.	n	d	m	p	$\sum Q_k$	$\sum q_{it}$
1	48	4	2	4	380	2189
2	96	4	3	4	600	2511
3	144	5	5	4	950	4026
4	192	5	6	4	1110	4971
5	240	4	8	4	1440	6871
6	288	6	9	4	1580	7074
7	72	3	3	6	600	3820
8	144	3	5	6	960	5968
9	216	3	7	6	1275	8098
10	288	4	11	6	1872	11,551
11	48	2	2	4	400	2189
12	96	3	3	4	598	2511
13	144	3	4	4	760	3738
14	192	4	6	4	1110	4971
15	240	5	9	4	1625	6871
16	288	5	9	4	1660	7074
17	72	3	3	6	600	3820
18	144	4	5	6	950	5968
19	216	3	7	6	1260	8098
20	288	5	11	6	1870	11,551

execution of the algorithm. The exploration and exploitation rates are observed to be influenced greatly by the selection of the parameters. Table 2 presents the parameter values set best attained for the proposed algorithm, which is obtained by trial and error method. Although the provided set need not be optimal, the algorithm is observed to perform much better with it as compared with other parameter values.

4.1 Discussion on numerical examples

In this section, a complete illustration of the obtained numerical results for the formulated MDMPVRP-TW as well as the proposed hybrid algorithm is presented through small as well as large instances discussed earlier. All the computational experiments are performed on a DELL PC (Intel Core i5 3.2 GHz) with 4 GB RAM and Windows 7 Operating System. The program codes are written in MATLAB R2017a.

In order to clearly delineate the vehicle route-schedule over the periods, a detailed solution of the first instance with 48 customer nodes is presented in table 3 and figure 1. The first column of the table shows the route number expressed in the form of an ordered pair, wherein the first element indicates the period number and the second one is the vehicle number. From this table, it can be clearly interpreted that every customer node from 1 to 12 is served in all the periods; each of customers 13 to 24 is served twice in the time horizon whereas each of the customers 25 to 48 is served only once. All the routes except (2, 2) and (3, 2) are observed to have demand that is higher than the vehicle capacity, thereby forcing them to visit an intermediate replenishment depot along their routes. Also every vehicle starts from the depot where it ends in the previous period. The scheduling plan of each vehicle over the periods is depicted in figure 1.

We have tried existing VNS algorithms to optimize the proposed MDMPVRP-TW but, perhaps due to high complexity, none of the algorithms got any success. However, performance of the hybrid VNS–TS algorithm with respect to the operators exchange and relocate is tested in each of the three ways: firstly, without the exchange operator; secondly, without the relocate operator and finally with

Table 2. Parameter set.

Parameters	Value
$\alpha_0, \beta_0, \gamma_0, \delta_0$	100
$\alpha_{min}, \beta_{min}, \gamma_{min}, \delta_{min}$	1
$\alpha_{max}, \beta_{max}, \gamma_{max}, \delta_{max}$	10,000
λ	0.5
ω	3–7
θ	10–30% of $ \mathcal{N}_k(S) $

Table 3. Optimal routes of vehicles for instance of 48 customers.

Route number	Route demand	Path
(1,1)	331	49 → 35 → 7 → 10 → 6 → 46 → 39 → 15 → 52 → 12 → 21 → 24 → 30 → 23 → 1 → 32 → 52
(1,2)	254	49 → 2 → 9 → 13 → 8 → 4 → 51 → 5 → 28 → 14 → 11 → 48 → 3 → 27 → 50
(2,1)	258	52 → 47 → 2 → 9 → 10 → 7 → 45 → 6 → 12 → 49 → 37 → 22 → 11 → 50
(2,2)	174	50 → 20 → 4 → 31 → 8 → 5 → 19 → 3 → 1 → 51
(3,1)	330	50 → 10 → 42 → 9 → 2 → 13 → 8 → 52 → 24 → 12 → 15 → 25 → 14 → 3 → 11 → 21 → 52
(3,2)	174	51 → 7 → 4 → 41 → 5 → 23 → 1 → 16 → 51
(4,1)	243	52 → 34 → 10 → 6 → 7 → 44 → 12 → 5 → 36 → 49 → 22 → 3 → 11 → 50
(4,2)	283	51 → 33 → 9 → 2 → 4 → 20 → 29 → 8 → 51 → 26 → 19 → 1 → 18 → 43 → 52

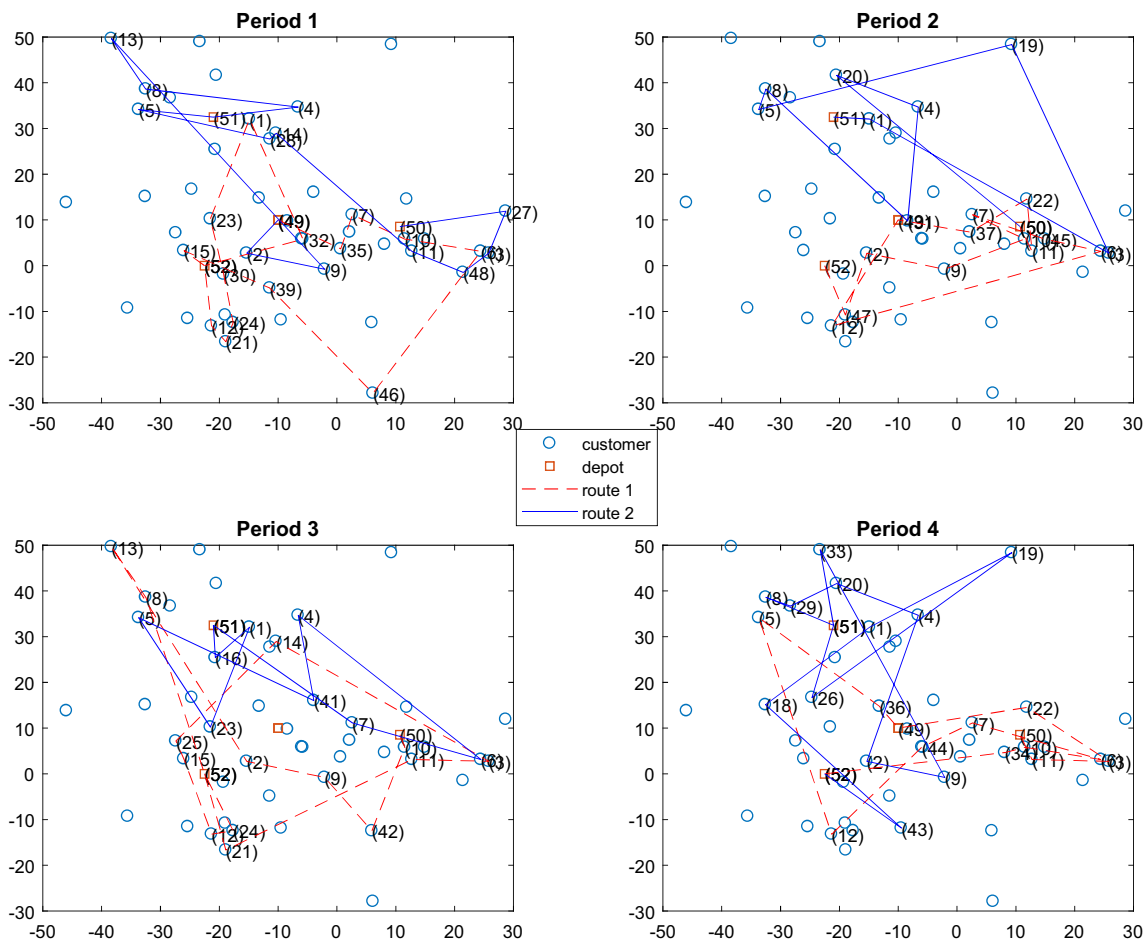


Figure 1. Optimal routes of vehicles over the periods.

Table 4. Comparisons between the operators and PSO.

Instance	Alg. w/o relocate				Alg. w/o exchange			
	Best	Avg	Worst	Avg run time (s)	Best	Avg	Worst	Avg run time (s)
1	1995.58	2125.766	2171.89	102.35	1981.23	2116.183	2166.37	99.69
2	3339.36	3653.642	3736.63	197.92	3459.11	3653.028	3711.25	199.01
3	5791.1	6225.031	6395.14	784.09	5899.19	6205.44	6377.21	802.9
4	6817.02	7085.194	7213.29	1079.93	6842.19	7108.421	7249.2	1082.53
5	7821.57	8414.596	8644.43	1849.06	7967.99	8397.833	8569.35	1881.32
6	10,037.67	10,449.417	10,591.4	3410.91	10,047.19	10,428.341	10,565.7	3493.11
7	5608.01	5911.822	6068.89	316.82	5596.87	5936.126	6092.04	319.99
8	8462.06	8867.578	9092.89	1237.72	8479.72	8837.522	8987.35	1272.06
9	13,471.09	13,961.7	14,289.65	2760.94	13,359.45	13,995.734	14,325.44	2620.1
10	17,841.43	18,690.79	19,077.28	3709.87	17,945.85	18,622.654	19,019.42	3680.27
11	1449.1	1599.267	1665.7	98.76	1459.95	1589.189	1649.97	94.66
12	2965.39	3124.651	3206.2	210.5	2926.54	3129.77	3211.2	234.03
13	4566.11	4825.058	4884.12	771.42	4615.11	4823.106	4880.44	759.08
14	4980.08	5318.054	5490.89	1042.42	4985.98	5293.614	5454.22	1121.23
15	4994.82	5420.935	5642.21	1816.57	5003.12	5398.004	5583.73	1801.59
16	6935.08	7514.79	7754.62	3289.6	7167.7	7509.599	7725.39	3362.51
17	3888.14	4120.824	4292.86	322	3896	4146.013	4328.92	328.88
18	5615.45	6010.835	6172	1147.22	5674.85	5988.384	6141.43	1126.47
19	8432.92	8889.943	9171.28	2606.11	8417.03	8935.17	9243.39	2530.07
20	10,198.69	10,772.468	11,042.17	3891.66	10,210.93	10,736.598	10,950.52	3845.87

Table 5. Comparisons between the operators and PSO.

Instance	Alg. w both				PSO			
	Best	Avg	Worst	Avg run time (s)	Best	Avg	Worst	Avg run time (s)
1	1930.26	2031.899	2105.78	121.85	1940.71	2075.351	2158.88	123.07
2	3321.46	3520.817	3647.41	241.37	3321.46	3575.66	3685.14	234.13
3	5773.53	5983.852	6128.64	933.44	5833.57	6063.372	6233.23	980.11
4	6599.45	6900.863	6993.52	1301.12	6753.58	6981.441	7133.75	1353.16
5	7774.33	8099.421	8264.14	2201.26	8040.12	8244.807	8489.61	2179.25
6	9992.6	10,194.375	10,305.73	3966.18	10,142.21	10,312.794	10,521.83	4164.49
7	5586.04	5717.365	5845.75	372.73	5608.01	5795.132	5967.72	376.46
8	8378.64	8593.135	8739.19	1473.48	8479.72	8698.391	8880.27	1429.28
9	13,264.6	13,648.92	13,806.13	3102.18	13,528.32	13,775.472	14,076.06	3195.25
10	17,841.43	18,122.446	18,254.41	4469.72	17,999.98	18,340.332	18,849.38	4425.02
11	1443.36	1499.283	1540.54	117.57	1466.23	1538.744	1620.47	114.04
12	2887.27	3024.331	3089.77	247.65	2887.27	3059.13	3154.77	260.03
13	4452.35	4684.118	4793.84	964.28	4615.11	4761.825	4865.36	925.71
14	4857.86	5133.273	5234.67	1271.24	5060.22	5206.405	5314.8	1245.82
15	4939.43	5170.296	5277.62	2242.68	5091.95	5269.953	5488.76	2354.81
16	6726.95	7248.013	7409.28	4061.23	7131.64	7365.984	7563.06	4142.45
17	3743.37	3959.511	4028.37	361.8	3813.21	4012.704	4149.84	365.42
18	5563.51	5800.723	5896.27	1433.75	5779.96	5896.161	6068.9	1405.08
19	8234.32	8563.62	8716.99	3018.1	8452.85	8694.44	8982.16	3048.28
20	10,062.24	10,524.043	10,664.2	4578.42	10,479.9	10,639.915	10,804.21	4807.34

both the exchange and relocate operators. Comparisons between the developed algorithm and the well-known PSO are also studied. The parameters involved in the proposed algorithm contain several random components as well.

Therefore for more accurate results, the algorithm is run for 10 times for 100 iterations in each run. The best obtained results of the conducted comparative study are then reported partly in table 4 and partly in table 5.

Comparisons from the tabulated results indicate that the best solution with minimum distance is obtained for the third case when the algorithm is tested with both the relocate and exchange operators in comparison with the other two situations. While being compared with PSO also, better outcomes are observed for the case of simultaneous incorporation of relocate and exchange operators in our proposed methodology.

5. Conclusion and future work

The aim of this paper is two-fold. First, to develop a more practical VRP that is mostly used in real life such as school bus routing, grocery distribution, oil and petroleum delivery, repairmen scheduling, etc. In this process, we propose a multi-depot multi-period vehicle routing problem with time window (MDMPVRP-TW) by considering allowable combination. A particular customer may appear along several routes and can be served by many vehicles over the period. However, all these routes and vehicle may not be feasible for the customer. Out of these routes and vehicles, we first select the feasible one and then make all possible combinations. Our objective is to find the most suitable combination for each customer in such a way that the total traversed distance or cost is minimum. The second objective of this paper is to propose a hybrid VNS–TS algorithm to optimize the mathematical model of MDMPVRP-TW. In this process, three operators for hybrid VNS–TS algorithm are considered in order to construct the neighbourhood structures for a solution. In order to find the initial solution, TS is modified and used. The efficiency and robustness are believed to be the strengths of the proposed algorithm. It is observed from the experimental results that the algorithm holds good for the proposed problem and is found to be preferably stable in obtaining the optimal solution.

Both the proposed model and algorithm can be further extended in several directions. One possible extension is that instead of a single visit to a customer by a vehicle within a period, multiple visits can be considered. Furthermore, soft TW in place of hard TW can also be considered as an extension of this work that will further strengthen the model because it will tackle an even more realistic situation.

List of symbols

Sets and indices

$V_1 = \{1, 2, \dots, n\}$	Set of customers
$V_2 = \{n + 1, \dots, n + d\}$	Set of depots
$V = V_1 \cup V_2$	Set of all nodes
i, j	Indices for the customer nodes over the set V

t	Index for the time period over the set $T = \{1, 2, \dots, p\}$
k	Index for heterogeneous vehicle over the set $K = \{1, 2, \dots, r\}$

Parameters

D_k	Maximum duration of vehicle k
Q_k	Capacity of the vehicle k
v	Average vehicle speed
r_i	Number of allowable combinations of i th customer
q_{it}	Demand at i th node in period t
d_{ij}	Distance between the nodes i and j
$[e_{it}, l_{it}]$	Time interval in which service must start at node i in period t
s_{ikt}	Service time required at i in t th period by k th vehicle
f_i	Visit frequency to the node i over entire planning horizon
C_i	Set of visit combinations of customer i
a_{rt}	1 if day t belongs to combination r , otherwise 0
Λ	$\max \{l_{it} i \in V_2\}$

Decision variables

ar_{ikt}	Arrival time of the vehicle k at i in t th period
y_{ikt}	Remaining cargo of k th vehicle on arrival at i in t th period
w_{ir}	1 if visit combination $r \in C_i$ is selected for node i , otherwise 0
E_{ikt}	1 if vehicle k visits depot i in period t for intermediate replenishment, otherwise 0
M_{ikt}	1 if vehicle k starts a tour from i th depot in period t , otherwise 0
N_{ikt}	1 if vehicle k ends a tour at i th depot in period t , otherwise 0
x_{ijkt}	1 if vehicle k goes from i to j in period t , otherwise 0

Acknowledgements

The authors would like to acknowledge the support provided by the Indian Institute of Technology Kharagpur through facilities for research. The first and third authors are grateful to Ministry of Human Resource Development for supporting their scientific studies with the Institute Research Assistantship.

References

- [1] Cordeau J F, Laporte G and Mercier A 2001 A unified tabu search heuristic for vehicle routing problems with time windows. *J. Oper. Res. Soc.* 52(8): 928–936
- [2] Yousefikhoshbakht M, Didehvar F and Rahmati F 2014 Solving the heterogeneous fixed fleet open vehicle routing problem by a combined metaheuristic algorithm. *Int. J. Prod. Res.* 52(9): 2565–2575

- [3] Taillard E, Badeau P, Gendreau M, Guertin F and Potvin J Y 1997 A tabu search heuristic for the vehicle routing problem with soft time windows. *Transp. Sci.* 31(2): 170–186
- [4] Paul A, Kumar R S, Rout C and Goswami A 2021 A bi-objective two-echelon pollution routing problem with simultaneous pickup and delivery under multiple time windows constraint. *OPSEARCH* 269(1): 1–32
- [5] Solomon M M 1987 Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.* 35(2): 254–265
- [6] Bräysy O and Gendreau M 2005 Vehicle routing problem with time windows, part I: route construction and local search algorithms. *Transp. Sci.* 39(1): 104–118
- [7] Chang T S, Wan Y W and Ooi W T 2009 A stochastic dynamic traveling salesman problem with hard time windows. *Eur. J. Oper. Res.* 198(3): 748–759
- [8] Miranda D M and Conceição S V 2016 The vehicle routing problem with hard time windows and stochastic travel and service time. *Expert Syst. Appl.* 64: 104–116
- [9] Chu J C, Yan S and Huang H J 2017 A multi-trip split-delivery vehicle routing problem with time windows for inventory replenishment under stochastic travel times. *Netw. Spatial Econ.* 17(1): 41–68
- [10] Hu C, Lu J, Liu X and Zhang G 2018 Robust vehicle routing problem with hard time windows under demand and travel time uncertainty. *Comput. Oper. Res.* 94: 139–153
- [11] Mancini S 2016 A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: formulation and adaptive large neighborhood search based matheuristic. *Transp. Res. Part C Emerg. Technol.* 70: 100–112
- [12] Dayarian I, Crainic T G, Gendreau M and Rei W 2015 A branch-and-price approach for a multi-period vehicle routing problem. *Comput. Oper. Res.* 55: 167–184
- [13] Luo Z, Qin H, Che C H and Lim A 2015 On service consistency in multi-period vehicle routing. *Eur. J. Oper. Res.* 243(3): 731–744
- [14] Dayarian I, Crainic T G, Gendreau M and Rei W 2016 An adaptive large-neighborhood search heuristic for a multi-period vehicle routing problem. *Transp. Res. Part E Logist. Transp. Rev.* 95: 95–123
- [15] Naccache S, Côté J F and Coelho L C 2018 The multi-pickup and delivery problem with time windows. *Eur. J. Oper. Res.* 269(1): 353–362
- [16] Lahyani R, Coelho L C and Renaud J 2018 Alternative formulations and improved bounds for the multi-depot fleet size and mix vehicle routing problem. *OR Spectrum* 40(1): 125–157
- [17] Michallet J, Prins C, Amodeo L, Yalaoui F and Vitry G 2014 Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services. *Comput. Oper. Res.* 41: 196–207
- [18] Tillman F A 1969 The multiple terminal delivery problem with probabilistic demands. *Transp. Sci.* 3(3): 192–204
- [19] Renaud J, Laporte G and Boctor F F 1996 A tabu search heuristic for the multi-depot vehicle routing problem. *Comput. Oper. Res.* 23(3): 229–235
- [20] Polacek M, Hartl R F, Doermer K and Reimann M 2004 A variable neighborhood search for the multi depot vehicle routing problem with time windows. *J. Heuristics* 10(6): 613–627
- [21] Crevier B, Cordeau J F and Laporte G 2007 The multi-depot vehicle routing problem with inter-depot routes. *Eur. J. Oper. Res.* 176(2): 756–773
- [22] Cordeau J F, Gendreau M and Laporte G 1997 A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Netw. Int. J.* 30(2): 105–119
- [23] Escobar J W, Linfati R, Toth P and Baldoquin M G 2014 A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. *J. Heuristics* 20(5): 483–509
- [24] Salhi S, Imran A and Wassan N A 2014 The multi-depot vehicle routing problem with heterogeneous vehicle fleet: formulation and a variable neighborhood search implementation. *Comput. Oper. Res.* 52: 315–325
- [25] Kumar R S, Kondapaneni K, Dixit V, Goswami A, Thakur L S and Tiwari M K 2016 Multi-objective modeling of production and pollution routing problem with time window: a self-learning particle swarm optimization approach. *Comput. Ind. Eng.* 99: 29–40
- [26] Soto M, Sevaux M, Rossi A and Reinholz A 2017 Multiple neighborhood search, tabu search and ejection chains for the multi-depot open vehicle routing problem. *Comput. Ind. Eng.* 107: 211–222
- [27] Zhou L, Baldacci R, Vigo D and Wang X 2018 A multi-depot two-echelon vehicle Routing problem with delivery options arising in the last mile Distribution. *Eur. J. Oper. Res.* 265(2): 765–778
- [28] Schneider M, Stenger A and Goetze D 2014 The electric vehicle-routing problem with time windows and recharging stations. *Transp. Sci.* 48(4): 500–520
- [29] Potvin J Y and Rousseau J M 1995 An exchange heuristic for routing problems with time windows. *J. Oper. Res. Soc.* 46(12): 1433–1446
- [30] Savelsbergh M W P 1992 The vehicle routing problem with time windows: minimizing route duration. *ORSA J. Comput.* 4(2): 146–154